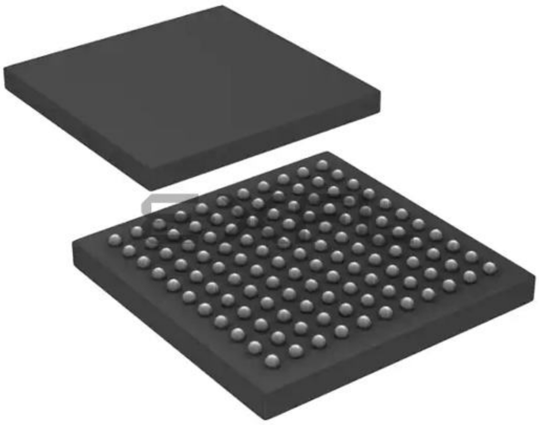


Welcome to E-XFL.COM

**What is "Embedded - Microcontrollers"?**

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

**Applications of "Embedded - Microcontrollers"**

| **Details** | |
|---|---|
| Product Status | Active |
| Core Processor | PIC |
| Core Size | 16-Bit |
| Speed | 32MHz |
| Connectivity | I²C, IrDA, LINbus, PMP, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT |
| Number of I/O | 85 |
| Program Memory Size | 512KB (170K x 24) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 32K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2V ~ 3.6V |
| Data Converters | A/D 24x10/12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 121-TFBGA |
| Supplier Device Package | 121-TFBGA (10x10) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic24fj512ga610-i-bg |

### 4.1.1 PROGRAM MEMORY ORGANIZATION

The program memory space is organized in word-addressable blocks. Although it is treated as 24 bits wide, it is more appropriate to think of each address of the program memory as a lower and upper word, with the upper byte of the upper word being unimplemented. The lower word always has an even address, while the upper word has an odd address (Figure 4-3).

Program memory addresses are always word-aligned on the lower word and addresses are incremented or decremented by two during code execution. This arrangement also provides compatibility with data memory space addressing and makes it possible to access data in the program memory space.

In Single Partition mode, user program memory is arranged in a contiguous block starting at address, 000000h.

### 4.1.2 DUAL PARTITION FLASH PROGRAM MEMORY ORGANIZATION

In the Dual Partition modes, the device's memory is divided evenly into two physical sections, known as Partition 1 and Partition 2. Each of these partitions contains its own program memory and Configuration Words. During program execution, the code on only one of these panels is executed; this is the Active Partition. The other partition, or the Inactive Partition, is not used, but can be programmed.

The Active Partition is always mapped to logical address, 000000h, while the Inactive Partition will always be mapped to logical address, 400000h. Note that even when the code partitions are switched between Active and Inactive by the user, the address of the Active Partition will still be at 000000h and the address of the Inactive Partition will still be at 400000h.

The Boot Sequence Configuration Word (FBTSEQ) determines whether Partition 1 or Partition 2 will be active after Reset. If the part is operating in Dual Partition mode, the partition with the lower Boot Sequence Number will operate as the Active Partition (FBTSEQ is unused in Single Partition mode). The partitions can be switched between Active and Inactive by reprogramming their Boot Sequence Numbers, but the Active Partition will not change until a device Reset is performed. If both Boot Sequence Numbers are the same, or if both are corrupted, the part will use Partition 1 as the Active Partition. If only one Boot Sequence Number is corrupted, the device will use the partition without a corrupted Boot Sequence Number as the Active Partition.

Should a Boot Sequence Number be invalid (or unprogrammed), it will be overridden to value, 0x000FFF (i.e., the highest possible Boot Sequence Number).

The user can also change which partition is active at run time using the `BOOTSWP` instruction. Issuing a `BOOTSWP` instruction does not affect which partition will be the Active Partition after a Reset. Figure 4-2 demonstrates how the relationship between Partitions 1 and 2, shown in red and blue respectively, and the Active and Inactive Partitions are affected by reprogramming the Boot Sequence Number or issuing a `BOOTSWP` instruction.

The P2ACTIV bit (NVMCON<10>) can be used to determine which physical partition is the Active Partition. If P2ACTIV = `1`, Partition 2 is active; if P2ACTIV = `0`, Partition 1 is active.

### 4.1.3 HARD MEMORY VECTORS

All PIC24F devices reserve the addresses between 000000h and 000200h for hard-coded program execution vectors. A hardware Reset vector is provided to redirect code execution from the default value of the PC on a device Reset to the actual start of code. A `GOTO` instruction is programmed by the user at 000000h, with the actual address for the start of code at 000002h.

The PIC24FJ1024GA610/GB610 devices can have up to two Interrupt Vector Tables (IVT). The first is located from addresses, 000004h to 0000FFh. The Alternate Interrupt Vector Table (AIVT) can be enabled by the AIVTDIS Configuration bit if the Boot Segment (BS) is present. If the user has configured a Boot Segment, the AIVT will be located at the address, ($\overline{\text{BSLIM<12:0>}}$ – 1) x 0x800. These vector tables allow each of the many device interrupt sources to be handled by separate ISRs. A more detailed discussion of the Interrupt Vector Tables is provided in **Section 8.1 "Interrupt Vector Table"**.

### 4.1.4 CONFIGURATION BITS OVERVIEW

The Configuration bits are stored in the last page location of implemented program memory. These bits can be set or cleared to select various device configurations. There are two types of Configuration bits: system operation bits and code-protect bits. The system operation bits determine the power-on settings for system-level components, such as the oscillator and the Watchdog Timer. The code-protect bits prevent program memory from being read and written.

Table 4-2 lists the Configuration register address range for each device in Single and Dual Partition modes. Table 4-2 lists all of the Configuration bits found in the PIC24FJ1024GA610/GB610 family devices, as well as their Configuration register locations. Refer to **Section 30.0 "Special Features"** in this data sheet for the full Configuration register description for each specific device.

**EXAMPLE 6-3: PROGRAMMING A DOUBLE WORD OF FLASH PROGRAM MEMORY (‘C’ LANGUAGE CODE)**

```
// C example using MPLAB XC16
unsigned long progAddr = 0xXXXXXX;          // Address of word to program
unsigned int  progData1L = 0xXXXX;          // Data to program lower word of word 1
unsigned char progData1H = 0xXX;            // Data to program upper byte of word 1
unsigned int  progData2L = 0xXXXX;          // Data to program lower word  of word 2
unsigned char progData2H = 0xXX;            // Data to program upper byte of word 2

//Set up NVMCON for word programming
NVMCON = 0x4001;                            // Initialize NVMCON
TBLPAG = 0xFA;                              // Point TBLPAG to the write latches

//Set up pointer to the first memory location to be written
NVMADRU = progAddr>>16;                     // Initialize PM Page Boundary SFR
NVMADR =  progAddr & 0xFFFF;                // Initialize lower word of address

//Perform TBLWT instructions to write latches
__builtin_tblwtl(0, progData1L);            // Write word 1 to address low word
__builtin_tblwth(0, progData2H);            // Write word 1 to upper byte
__builtin_tblwtl(1, progData2L);            // Write word 2 to address low word
__builtin_tblwth(1, progData2H);            // Write word 2 to upper byte
asm(“DISI #5”);                             // Block interrupts with priority <7 for next 5
                                            // instructions
__builtin_write_NVM();                      // XC16 function to perform unlock sequence and set WR
```

**REGISTER 9-5: DCOCON: DIGITALLY CONTROLLED OSCILLATOR ENABLE REGISTER**

| U-0 | U-0 | R/W-0 | U-0 | R/W-0 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| | | DCOEN | | DCOFSEL3 | DCOFSEL2 | DCOFSEL1 | DCOFSEL0 |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13 **DCOEN:** DCO Enable bit
1 = DCO continues to operate during Sleep mode
0 = DCO is inactive during Sleep mode

bit 12 **Unimplemented:** Read as ‘0’

bit 11-8 **DCOFSEL<3:0>:** DCO Frequency Select bits
0000 = 1 MHz
0001 = 2 MHz
0010 = 3 MHz
0011 = 4 MHz
0100 = 5 MHz
0101 = 6 MHz
0110 = 7 MHz
0111 = 8 MHz (most accurate oscillator setting)
1000 = Reserved; do not use
1001 = Reserved; do not use
1010 = Reserved; do not use
1011 = Reserved; do not use
1100 = Reserved; do not use
1101 = Reserved; do not use
1110 = 15 MHz
1111 = 30 MHz

bit 7-0 **Unimplemented:** Read as ‘0’

### REGISTER 10-8: PMD8: PERIPHERAL MODULE DISABLE REGISTER 8

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| U6MD | U5MD | CLC4MD | CLC3MD | CLC2MD | CLC1MD | — | — |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7 **U6MD:** UART6 Module Disable bit
1 = Module is disabled
0 = Module power and clock sources are enabled

bit 6 **U5MD:** UART5 Module Disable bit
1 = Module is disabled
0 = Module power and clock sources are enabled

bit 5 **CLC4MD:** CLC4 Module Disable bit
1 = Module is disabled
0 = Module power and clock sources are enabled

bit 4 **CLC3MD:** CLC3 Module Disable bit
1 = Module is disabled
0 = Module power and clock sources are enabled

bit 3 **CLC2MD:** CLC2 Module Disable bit
1 = Module is disabled
0 = Module power and clock sources are enabled

bit 2 **CLC1MD:** CLC1 Module Disable bit
1 = Module is disabled
0 = Module power and clock sources are enabled

bit 1-0 **Unimplemented:** Read as ‘0’

## 11.0 I/O PORTS

> **Note:** This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"I/O Ports with Peripheral Pin Select (PPS)"** (DS39711), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

All of the device pins (except $V_{DD}$, $V_{SS}$, $\overline{MCLR}$ and OSCI/CLKI) are shared between the peripherals and the Parallel I/O (PIO) ports. All I/O input ports feature Schmitt Trigger (ST) inputs for improved noise immunity.

### 11.1 Parallel I/O (PIO) Ports

A Parallel I/O port that shares a pin with a peripheral is, in general, subservient to the peripheral. The peripheral's output buffer data and control signals are provided to a pair of multiplexers. The multiplexers select whether the peripheral or the associated port has ownership of the output data and control signals of the I/O pin. The logic also prevents "loop through", in which a port's digital output can drive the input of a peripheral that shares the same pin. Figure 11-1 shows how ports are shared with other peripherals and the associated I/O pin to which they are connected.

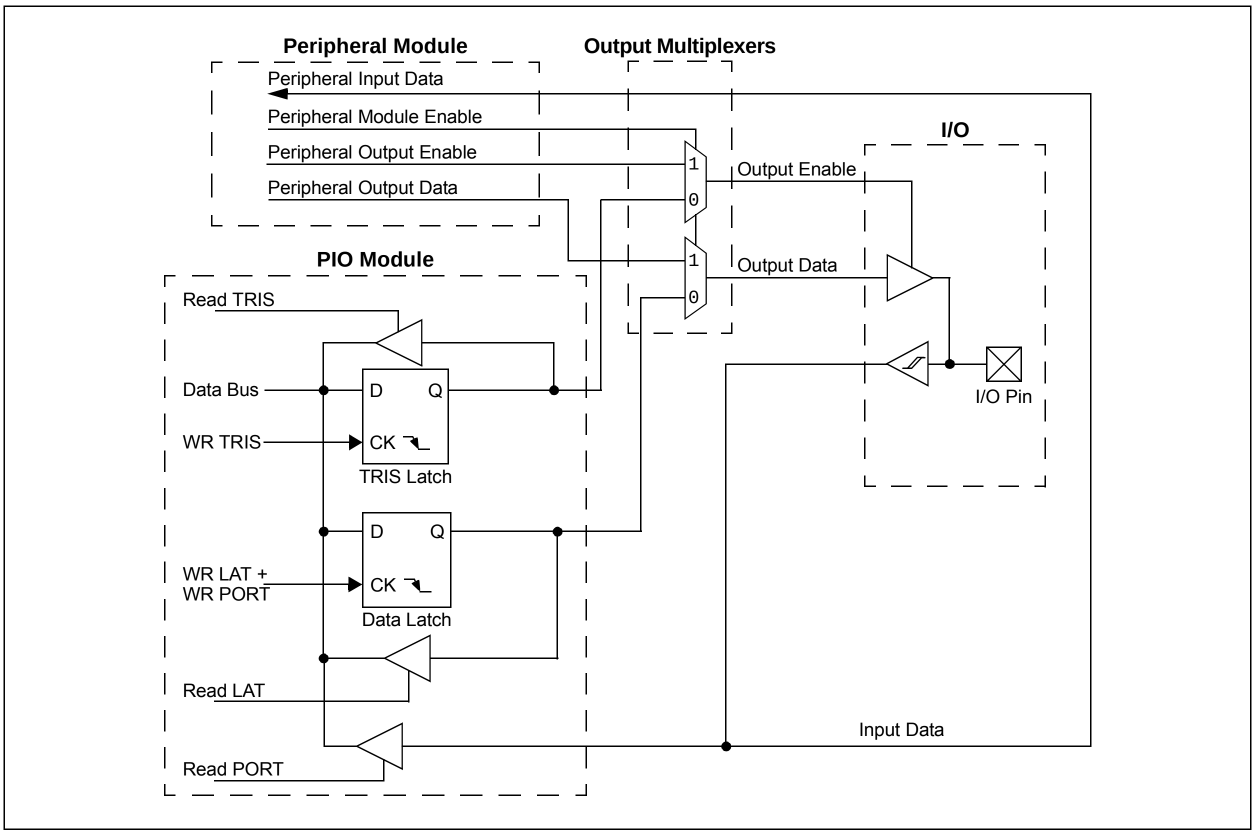
When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin may be read, but the output driver for the parallel port bit will be disabled. If a peripheral is enabled, but the peripheral is not actively driving a pin, that pin may be driven by a port.

All port pins have three registers directly associated with their operation as digital I/Os and one register associated with their operation as analog inputs. The Data Direction register (TRISx) determines whether the pin is an input or an output. If the data direction bit is a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Reads from the Output Latch register (LATx), read the latch; writes to the latch, write the latch. Reads from the PORTx register, read the port pins; writes to the port pins, write the latch.

Any bit and its associated data and control registers that are not valid for a particular device will be disabled. That means the corresponding LATx and TRISx registers, and the port pin, will read as zeros.

When a pin is shared with another peripheral or function that is defined as an input only, it is regarded as a dedicated port because there is no other competing source of inputs.

**FIGURE 11-1: BLOCK DIAGRAM OF A TYPICAL SHARED PORT STRUCTURE**



#### 11.4.3.3 Mapping Limitations

The control schema of the Peripheral Pin Select is extremely flexible. Other than systematic blocks that prevent signal contention, caused by two physical pins being configured as the same functional input or two functional outputs configured as the same pin, there are no hardware enforced lockouts. The flexibility extends to the point of allowing a single input to drive multiple peripherals or a single functional output to drive multiple output pins.

#### 11.4.3.4 Mapping Exceptions for PIC24FJ1024GA610/GB610 Family Devices

Although the PPS registers theoretically allow for inputs to be remapped to up to 64 pins, or for outputs to be remapped from 32 pins, not all of these are implemented in all devices. For 100-pin or 121-pin variants of the PIC24FJ1024GA610/GB610 family devices, 32 remappable input/output pins are available and 12 remappable input pins are available. For 64-pin variants, 29 input/outputs and 1 input are available. The differences in available remappable pins are summarized in Table 11-5.

When developing applications that use remappable pins, users should also keep these things in mind:

- For the RPINRx registers, bit combinations corresponding to an unimplemented pin for a particular device are treated as invalid; the corresponding module will not have an input mapped to it.
- For RPORx registers, the bit fields corresponding to an unimplemented pin will also be unimplemented; writing to these fields will have no effect.

### 11.4.4 CONTROLLING CONFIGURATION CHANGES

Because peripheral remapping can be changed during run time, some restrictions on peripheral remapping are needed to prevent accidental configuration changes. PIC24F devices include three features to prevent alterations to the peripheral map:

- Control register lock sequence
- Continuous state monitoring
- Configuration bit remapping lock

#### 11.4.4.1 Control Register Lock

Under normal operation, writes to the RPINRx and RPORx registers are not allowed. Attempted writes will appear to execute normally, but the contents of the registers will remain unchanged. To change these registers, they must be unlocked in hardware. The register lock is controlled by the IOLOCK bit (OSCCON<6>). Setting IOLOCK prevents writes to the control registers; clearing IOLOCK allows writes.

To set or clear IOLOCK, a specific command sequence must be executed:

1. Write 46h to OSCCON<7:0>.
2. Write 57h to OSCCON<7:0>.
3. Clear (or set) IOLOCK as a single operation.

Unlike the similar sequence with the oscillator's LOCK bit, IOLOCK remains in one state until changed. This allows all of the Peripheral Pin Selects to be configured with a single unlock sequence, followed by an update to all control registers, then locked with a second lock sequence.

#### 11.4.4.2 Continuous State Monitoring

In addition to being protected from direct writes, the contents of the RPINRx and RPORx registers are constantly monitored in hardware by shadow registers. If an unexpected change in any of the registers occurs (such as cell disturbances caused by ESD or other external events), a Configuration Mismatch Reset will be triggered.

#### 11.4.4.3 Configuration Bit Pin Select Lock

As an additional level of safety, the device can be configured to prevent more than one write session to the RPINRx and RPORx registers. The IOL1WAY (FOSC<5>) Configuration bit blocks the IOLOCK bit from being cleared after it has been set once. If IOLOCK remains set, the register unlock procedure will not execute and the Peripheral Pin Select Control registers cannot be written to. The only way to clear the bit and re-enable peripheral remapping is to perform a device Reset.

In the default (unprogrammed) state, IOL1WAY is set, restricting users to one write session. Programming IOL1WAY allows users unlimited access (with the proper use of the unlock sequence) to the Peripheral Pin Select registers.

**TABLE 11-5: REMAPPABLE PIN EXCEPTIONS FOR PIC24FJ1024GA610/GB610 FAMILY DEVICES**

| Device | RPn Pins (I/O) | | RPIn Pins | |
|---|---|---|---|---|
| | Total | Unimplemented | Total | Unimplemented |
| PIC24FJXXXGB606 | 28 | RP5, RP15, RP30, RP31 | 1 | All except RPI37 |
| PIC24FJXXXGX61X | 32 | — | 12 | — |
| PIC24FJXXXGA606 | 29 | RP5, RP15, RP31 | 1 | All except RPI37 |

### REGISTER 11-20: RPINR8: PERIPHERAL PIN SELECT INPUT REGISTER 8

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | — | IC3R5 | IC3R4 | IC3R3 | IC3R2 | IC3R1 | IC3R0 |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **IC3R<5:0>:** Assign Input Capture 3 (IC3) to Corresponding RPn or RPIn Pin bits

### REGISTER 11-21: RPINR11: PERIPHERAL PIN SELECT INPUT REGISTER 11

| U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | — | OCFBR5 | OCFBR4 | OCFBR3 | OCFBR2 | OCFBR1 | OCFBR0 |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | — | OCFAR5 | OCFAR4 | OCFAR3 | OCFAR2 | OCFAR1 | OCFAR0 |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **OCFBR<5:0>:** Assign Output Compare Fault B (OCFB) to Corresponding RPn or RPIn Pin bits

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **OCFAR<5:0>:** Assign Output Compare Fault A (OCFA) to Corresponding RPn or RPIn Pin bits

# 14.0 INPUT CAPTURE WITH DEDICATED TIMERS

| Note: | This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"Input Capture with Dedicated Timer"** (DS70000352), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM. |
|---|---|

Devices in the PIC24FJ1024GA610/GB610 family contain six independent input capture modules. Each of the modules offers a wide range of configuration and operating options for capturing external pulse events and generating interrupts.

Key features of the input capture module include:

- Hardware-Configurable for 32-Bit Operation in all modes by Cascading Two Adjacent modules
- Synchronous and Trigger modes of Output Compare Operation with up to 31 User-Selectable Sync/Trigger Sources Available
- A 4-Level FIFO Buffer for Capturing and Holding Timer Values for Several Events
- Configurable Interrupt Generation
- Up to 6 Clock Sources Available for each module, Driving a Separate Internal 16-Bit Counter

The module is controlled through two registers: ICxCON1 (Register 14-1) and ICxCON2 (Register 14-2). A general block diagram of the module is shown in Figure 14-1.

## 14.1 General Operating Modes

### 14.1.1 SYNCHRONOUS AND TRIGGER MODES

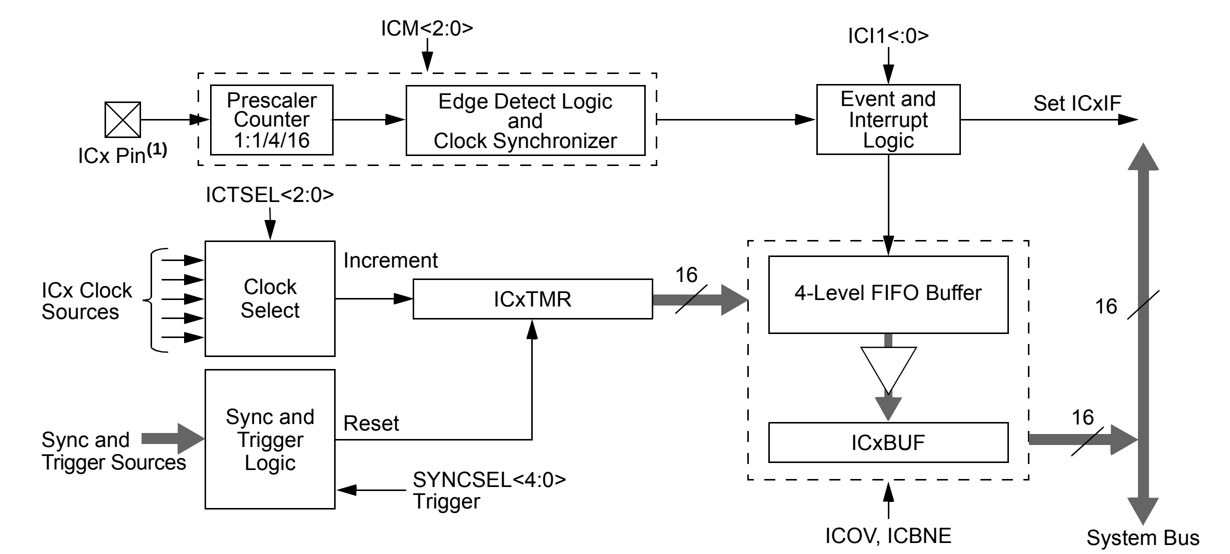
When the input capture module operates in a Free-Running mode, the internal 16-bit counter, ICxTMR, counts up continuously, wrapping around from FFFFh to 0000h on each overflow. Its period is synchronized to the selected external clock source. When a capture event occurs, the current 16-bit value of the internal counter is written to the FIFO buffer.

In Synchronous mode, the module begins capturing events on the ICx pin as soon as its selected clock source is enabled. Whenever an event occurs on the selected Sync source, the internal counter is reset. In Trigger mode, the module waits for a Sync event from another internal module to occur before allowing the internal counter to run.

Standard, free-running operation is selected by setting the SYNCSEL<4:0> bits (ICxCON2<4:0>) to '00000' and clearing the ICTRIG bit (ICxCON2<7>). Synchronous and Trigger modes are selected any time the SYNCSELx bits are set to any value except '00000'. The ICTRIG bit selects either Synchronous or Trigger mode; setting the bit selects Trigger mode operation. In both modes, the SYNCSELx bits determine the Sync/Trigger source.

When the SYNCSELx bits are set to '00000' and ICTRIG is set, the module operates in Software Trigger mode. In this case, capture operations are started by manually setting the TRIGSTAT bit (ICxCON2<6>).

**FIGURE 14-1: INPUT CAPTURE x BLOCK DIAGRAM**




**Note 1:** The ICx input must be assigned to an available RPn/RPIn pin before use. See **Section 11.4 "Peripheral Pin Select (PPS)"** for more information.

## 16.4 Input Capture Mode

Input Capture mode is used to capture a timer value from an independent timer base upon an event on an input pin or other internal Trigger source. The input capture features are useful in applications requiring frequency (time period) and pulse measurement. Figure 16-6 depicts a simplified block diagram of the Input Capture mode.

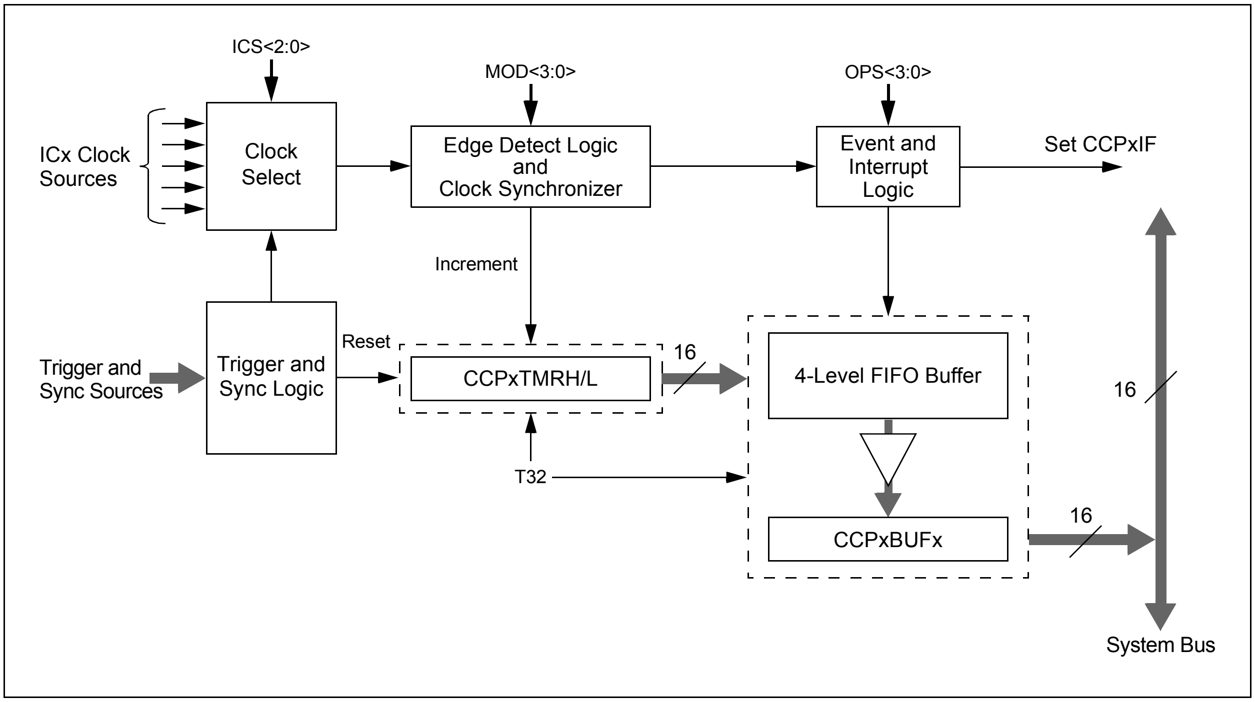
Input Capture mode uses a dedicated 16/32-bit, synchronous, up counting timer for the capture function. The timer value is written to the FIFO when a capture event occurs. The internal value may be read (with a synchronization delay) using the CCPxTMRH/L registers.

To use Input Capture mode, the CCSEL bit (CCPxCON1L<4>) must be set. The T32 and MOD<3:0> bits are used to select the proper Capture mode, as shown in Table 16-3.

**TABLE 16-3: INPUT CAPTURE MODES**

| MOD<3:0> (CCPxCON1L<3:0>) | T32 (CCPxCON1L<5>) | Operating Mode |
|---|---|---|
| 0000 | 0 | Edge Detect (16-bit capture) |
| 0000 | 1 | Edge Detect (32-bit capture) |
| 0001 | 0 | Every Rising (16-bit capture) |
| 0001 | 1 | Every Rising (32-bit capture) |
| 0010 | 0 | Every Falling (16-bit capture) |
| 0010 | 1 | Every Falling (32-bit capture) |
| 0011 | 0 | Every Rise/Fall (16-bit capture) |
| 0011 | 1 | Every Rise/Fall (32-bit capture) |
| 0100 | 0 | Every 4th Rising (16-bit capture) |
| 0100 | 1 | Every 4th Rising (32-bit capture) |
| 0101 | 0 | Every 16th Rising (16-bit capture) |
| 0101 | 1 | Every 16th Rising (32-bit capture) |

**FIGURE 16-6: INPUT CAPTURE x BLOCK DIAGRAM**



## REGISTER 22-13: ALMDATEL: RTCC ALARM DATE REGISTER (LOW)

| U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | DAYTEN1 | DAYTEN0 | DAYONE3 | DAYONE2 | DAYONE1 | DAYONE0 |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | WDAY2 | WDAY1 | WDAY0 |
| bit 7 | | | | | | | bit 0 |

| Legend: | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13-12 **DAYTEN<1:0>:** Binary Coded Decimal Value of Days ‘10’ Digit bits
Contains a value from 0 to 3.

bit 11-8 **DAYONE<3:0>:** Binary Coded Decimal Value of Days ‘1’ Digit bits
Contains a value from 0 to 9.

bit 7-3 **Unimplemented:** Read as ‘0’

bit 2-0 **WDAY<2:0>:** Binary Coded Decimal Value of Weekdays ‘1’ Digit bits
Contains a value from 0 to 6.

## REGISTER 22-14: ALMDATEH: RTCC ALARM DATE REGISTER (HIGH)

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| YRTEN3 | YRTEN2 | YRTEN1 | YRTEN0 | YRONE3 | YRONE2 | YRONE1 | YRONE0 |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | MTHTEN | MTHONE3 | MTHONE2 | MTHONE1 | MTHONE0 |
| bit 7 | | | | | | | bit 0 |

| Legend: | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-12 **YRTEN<3:0>:** Binary Coded Decimal Value of Years ‘10’ Digit bits

bit 11-8 **YRONE<3:0>:** Binary Coded Decimal Value of Years ‘1’ Digit bits

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4 **MTHTEN:** Binary Coded Decimal Value of Months ‘10’ Digit bit
Contains a value from 0 to 1.

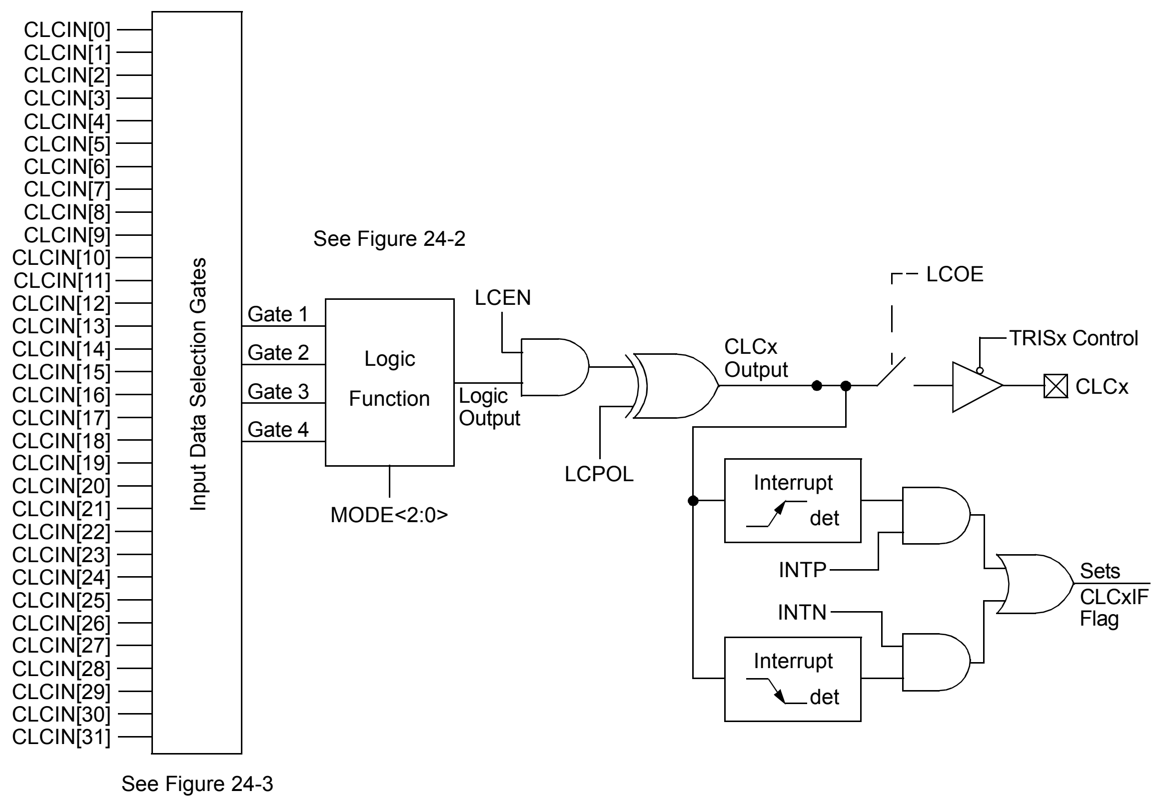
bit 3-0 **MTHONE<3:0>:** Binary Coded Decimal Value of Months ‘1’ Digit bits
Contains a value from 0 to 9.

## 24.0 CONFIGURABLE LOGIC CELL (CLC)

The Configurable Logic Cell (CLC) module allows the user to specify combinations of signals as inputs to a logic function and to use the logic output to control other peripherals or I/O pins. This provides greater flexibility and potential in embedded designs, since the CLC module can operate outside the limitations of software execution and supports a vast amount of output designs.

There are four input gates to the selected logic function. These four input gates select from a pool of up to 32 signals that are selected using four data source selection multiplexers. Figure 24-1 shows an overview of the module. Figure 24-3 shows the details of the data source multiplexers and logic input gate connections.

**FIGURE 24-1: CLCx MODULE**




**Note:** All register bits shown in this figure can be found in the CLCxCONL register.

## 25.2 Extended DMA Operations

In addition to the standard features available on all 12-bit A/D Converters, PIC24FJ1024GA610/GB610 family devices implement a limited extension of DMA functionality. This extension adds features that work with the device's DMA Controller to expand the A/D module's data storage abilities beyond the module's built-in buffer.

The Extended DMA functionality is controlled by the DMAEN bit (AD1CON1<11>); setting this bit enables the functionality. The DMABM bit (AD1CON1<12>) configures how the DMA feature operates.

### 25.2.1 EXTENDED BUFFER MODE

Extended Buffer mode (DMABM = 1) maps the A/D Data Buffer registers and data from all channels above 26 into a user-specified area of data RAM. This allows users to read the conversion results of channels above 26, which do not have their own memory-mapped A/D buffer locations, from data memory.

To accomplish this, the DMA must be configured in Peripheral Indirect Addressing mode and the DMA destination address must point to the beginning of the buffer. The DMA count must be set to generate an interrupt after the desired number of conversions.

In Extended Buffer mode, the A/D control bits will function similarly to non-DMA modes. The BUFREGEN bit will still select between FIFO mode and Channel-Aligned mode, but the number of words in the destination FIFO will be determined by the SMPI<4:0> bits in DMA mode. In FIFO mode, the BUFM bit will still split the output FIFO into two sets of 13 results (the SMPIx bits should be set accordingly), and the BUFS bit will still indicate which set of results is being written to and which can be read.

### 25.2.2 PIA MODE

When DMABM = 0, the A/D module is configured to function with the DMA Controller for Peripheral Indirect Addressing (PIA) mode operations. In this mode, the A/D module generates an 11-bit Indirect Address (IA). This is ORed with the destination address in the DMA Controller to define where the A/D conversion data will be stored.

In PIA mode, the buffer space is created as a series of contiguous smaller buffers, one per analog channel. The size of the channel buffer determines how many analog channels can be accommodated. The size of the buffer is selected by the DMABL<2:0> bits (AD1CON4<2:0>). The size options range from a single word per buffer to 128 words. Each channel is allocated a buffer of this size, regardless of whether or not the channel will actually have conversion data.

The IA is created by combining the base address within a channel buffer with three to five bits (depending on the buffer size) to identify the channel. The base address ranges from zero to seven bits wide, depending on the buffer size. The address is right-padded with a '0' in order to maintain address alignment in the Data Space. The concatenated channel and base address bits are then left-padded with zeros, as necessary, to complete the 11-bit IA.

The IA is configured to auto-increment which channel is written in each analog input's sub-buffer during write operations by using the SMPIx bits (AD1CON2<6:2>).

As with PIA operations for any DMA-enabled module, the base destination address in the DMADSTn register must be masked properly to accommodate the IA. Table 25-1 shows how complete addresses are formed. Note that the address masking varies for each buffer size option. Because of masking requirements, some address ranges may not be available for certain buffer sizes. Users should verify that the DMA base address is compatible with the buffer size selected.

Figure 25-2 shows how the parts of the address define the buffer locations in data memory. In this case, the module "allocates" 256 bytes of data RAM (1000h to 1100h) for 32 buffers of four words each. However, this is not a hard allocation and nothing prevents these locations from being used for other purposes. For example, in the current case, if Analog Channels 1, 3 and 8 are being sampled and converted, conversion data will only be written to the channel buffers, starting at 1008h, 1018h and 1040h. The holes in the PIA buffer space can be used for any other purpose. It is the user's responsibility to keep track of buffer locations and prevent data overwrites.

## REGISTER 25-7: ANCFG: A/D BAND GAP REFERENCE CONFIGURATION REGISTER

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | VBGUSB(1) | VBGADC(1) | VBGCMP(1) | VBGEN(1) |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3 **VBGUSB:** Band Gap Reference Enable for USB bit(1)
1 = Band gap reference is enabled
0 = Band gap reference is disabled

bit 2 **VBGADC:** Band Gap Reference Enable for A/D bit(1)
1 = Band gap reference is enabled
0 = Band gap reference is disabled

bit 1 **VBGCMP:** Band Gap Reference Enable for CTMU and Comparator bit(1)
1 = Band gap reference is enabled
0 = Band gap reference is disabled

bit 0 **VBGEN:** Band Gap Reference Enable for VREG, BOR, HLVD, FRC, DCO, NVM and A/D Boost bit(1)
1 = Band gap reference is enabled
0 = Band gap reference is disabled

**Note 1:** When a module requests a band gap reference voltage, that reference will be enabled automatically after a brief start-up time. The user can manually enable the band gap references using the ANCFG register before enabling the module requesting the band gap reference to avoid this startup time (~1 ms).

### REGISTER 25-10: AD1CSSH: A/D INPUT SCAN SELECT REGISTER (HIGH WORD)

| U-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | CSS<30:28> | | | — | CSS<26:24> | | |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| CSS<23:16> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

| Legend: | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15 **Unimplemented:** Read as ‘0’

bit 14-12 **CSS<30:28>:** A/D Input Scan Selection bits
1 = Includes corresponding channel for input scan
0 = Skips channel for input scan

bit 11 **Unimplemented:** Read as ‘0’

bit 10-0 **CSS<26:16>:** A/D Input Scan Selection bits
1 = Includes corresponding channel for input scan
0 = Skips channel for input scan

### REGISTER 25-11: AD1CSSL: A/D INPUT SCAN SELECT REGISTER (LOW WORD)

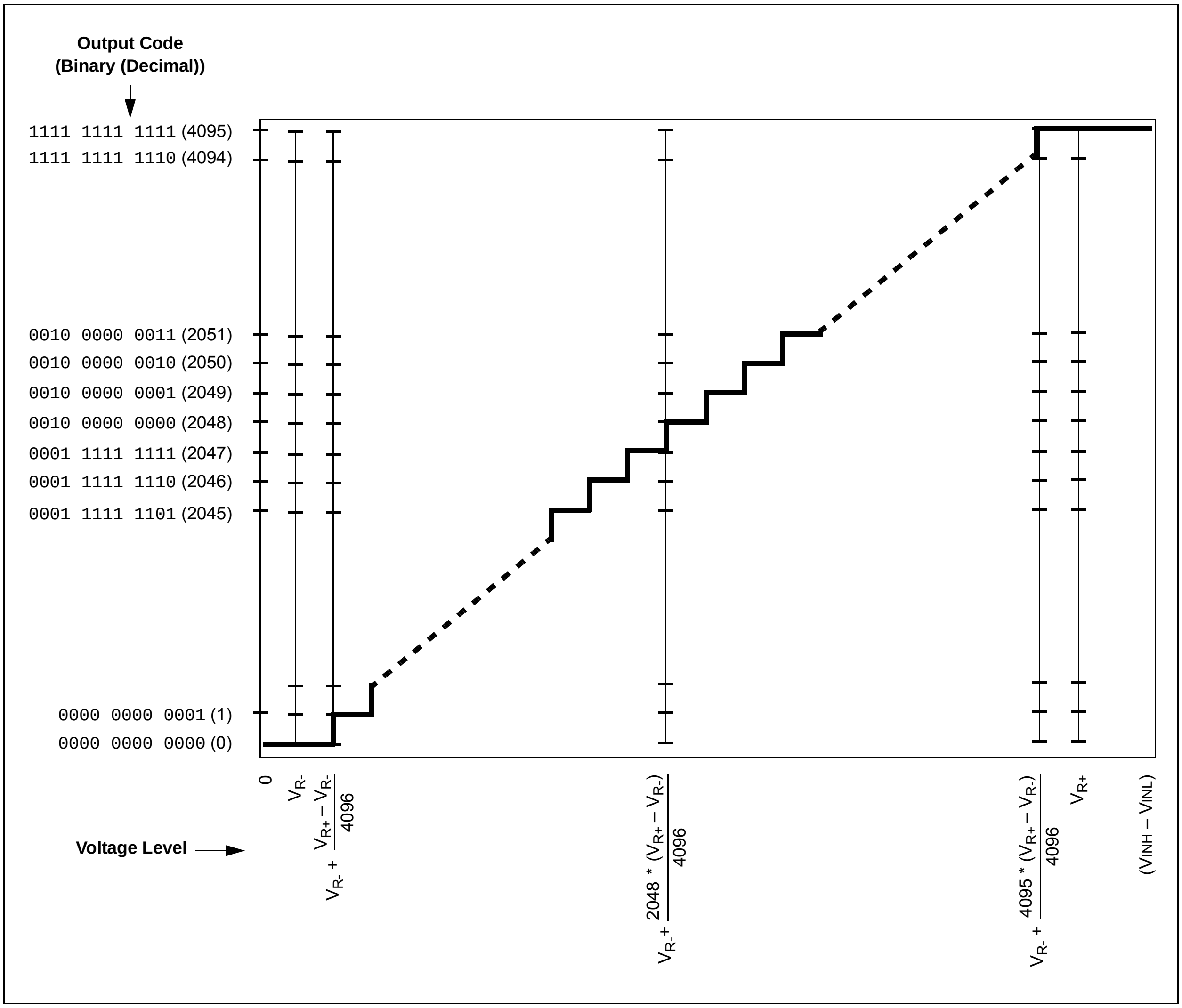
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| CSS<15:8> | | | | | | | |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| CSS<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

| Legend: | | | |
|---|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |
| -n = Value at POR | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15-0 **CSS<15:0>:** A/D Input Scan Selection bits
1 = Includes corresponding channel for input scan
0 = Skips channel for input scan

**FIGURE 25-4: 12-BIT A/D TRANSFER FUNCTION**



**NOTES:**

**NOTES:**

### REGISTER 29-1: HLVDCON: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER

| R/W-0 | U-0 | R/W-0 | U-0 | R/W-0 | r-1 | r-1 | R-0, HS, HC |
|---|---|---|---|---|---|---|---|
| HLVDEN | — | LSIDL | — | VDIR | BGVST | IRVST | LVDEVT(2) |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | HLVDL3 | HLVDL2 | HLVDL1 | HLVDL0 |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | HS = Hardware Settable bit | HC = Hardware Clearable bit | r = Reserved bit |
|---|---|---|---|
| R = Readable bit | W = Writable bit | ‘0’ = Bit is cleared | x = Bit is unknown |
| -n = Value at POR | ‘1’ = Bit is set | U = Unimplemented bit, read as ‘0’ | |

bit 15 **HLVDEN:** High/Low-Voltage Detect Power Enable bit
1 = HLVD is enabled
0 = HLVD is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **LSIDL:** HLVD Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **Unimplemented:** Read as ‘0’

bit 11 **VDIR:** Voltage Change Direction Select bit
1 = Event occurs when voltage equals or exceeds trip point (HLVDL<3:0>)
0 = Event occurs when voltage equals or falls below trip point (HLVDL<3:0>)

bit 10 **BGVST:** Reserved bit (value is always ‘1’)

bit 9 **IRVST:** Reserved bit (value is always ‘1’)

bit 8 **LVDEVT:** Low-Voltage Event Status bit(2)
1 = LVD event is true during current instruction cycle
0 = LVD event is not true during current instruction cycle

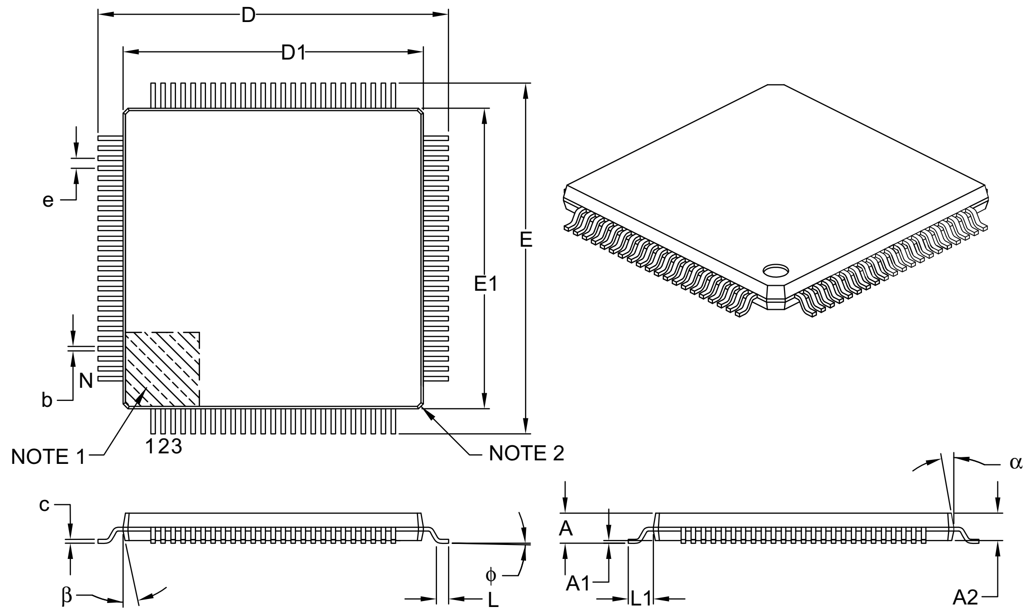
bit 7-4 **Unimplemented:** Read as ‘0’

bit 3-0 **HLVDL<3:0>:** High/Low-Voltage Detection Limit bits
1111 = External analog input is used (input comes from the HLVDIN pin)
1110 = Trip Point 1(1)
1101 = Trip Point 2(1)
1100 = Trip Point 3(1)
•
•
•
0100 = Trip Point 11(1)
00xx = Unused

**Note 1:** For the actual trip point, see **Section 33.0 “Electrical Characteristics”**.
**2:** The LVDIF flag cannot be cleared by software unless LVDEVT = 0. The voltage must be monitored so that the HLVD condition (as set by VDIR and HLVDL<3:0>) is not asserted.

## 100-Lead Plastic Thin Quad Flatpack (PT) – 12x12x1 mm Body, 2.00 mm [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 100 | |
| Lead Pitch | e | | 0.40 BSC | |
| Overall Height | A | – | – | 1.20 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Standoff | A1 | 0.05 | – | 0.15 |
| Foot Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | φ | 0° | 3.5° | 7° |
| Overall Width | E | | 14.00 BSC | |
| Overall Length | D | | 14.00 BSC | |
| Molded Package Width | E1 | | 12.00 BSC | |
| Molded Package Length | D1 | | 12.00 BSC | |
| Lead Thickness | c | 0.09 | – | 0.20 |
| Lead Width | b | 0.13 | 0.18 | 0.23 |
| Mold Draft Angle Top | α | 11° | 12° | 13° |
| Mold Draft Angle Bottom | β | 11° | 12° | 13° |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-100B